Thesis Work

Pegah Rahimian

2019
1. NormalSentiment, a novel contextual Semantic Approach for Sentiment Analysis and Text Mining

Pegah Rahimian
Msc in Business Information Systems
2019
Table of Content:

Contents

1. NormalSentiment, a novel contextual Semantic Approach for Sentiment Analysis and Text Mining ................................................................. 2
2. Introduction .................................................................................................................. 7
3. Literature Review ......................................................................................................... 10
   3.1. Sentiment analysis in text ...................................................................................... 10
   3.1.1. Machine learning approach .............................................................................. 10
   3.1.2. Lexicon-based approach .................................................................................. 11
   3.1.2.1. Statistical approach ...................................................................................... 11
   3.1.2.2. Semantic approach ...................................................................................... 12
3.2. Stock market prediction with SA ........................................................................... 12
3.3. Sentiment lexicons .................................................................................................. 13
4. Theoretical Framework ............................................................................................. 14
   4.1. Assumptions .......................................................................................................... 15
   4.2. NormalSentiment approach ................................................................................ 16
   4.2.1. The position of words in the polar coordinate systems .................................... 16
   4.2.2. Axis labelling in two sentiment regions ............................................................. 17
   4.2.3. Normal sentiment distribution for the sentiment of the term "V" ....................... 18
   4.2.4. Normal distribution properties for sentiment analysis .................................... 19
5. Methodology .............................................................................................................. 22
6. Elaboration of the central topic .................................................................................. 23
   6.1. First Approach: Data Annotation with R package: .............................................. 24
   6.2. Second Approach: NormalSentiment Approach (Contextual Semantics): ............ 31
   6.2.1. Using a machine learning approach: ............................................................... 31
   6.2.2. Data annotation: ............................................................................................ 36
   6.2.2.1. NormalSentiment for iPhone ......................................................................... 37
   6.2.2.2. NormalSentiment for Galaxy ....................................................................... 38
   6.2.2.3. Confidence interval for expectation (Average) : ............................................ 41
7. Analysis and Results .............................................................................................................43
7.1. Baselines: .........................................................................................................................43
7.2. Experimental Results.......................................................................................................45
7.2.1. NormalSentiment Method Performance ......................................................................45
7.2.2. Comparison of NormalSentiment results with other approaches .........................48
7.2.3 Comparison of Performance (NormalSentiment vs. Senti-Median) .........................51
8. Conclusion..........................................................................................................................52
9. References..........................................................................................................................54
2. Introduction

Opinion mining can be considered as an application of natural language processing (NLP) in order to analyse how people or customers are feeling about a specific product or brand. Opinion mining can be a system which collects and classifies customers’ opinions regarding different products and brands. Seeking the automation purpose, opinion mining can use Machine Learning which is a type of artificial intelligence (AI) that will be able to automatically analyse data for getting sentiment from text. Opinion mining can be applicable and useful in several aspects. It can help measuring how much successful a campaign or new product development can be. For example, web mining can show how much functional a product like a camera can be, but that is too much heavy. This kind of systematic approach to get public opinion, will help the vendors to understand a clear picture of customers’ taste, much better than surveys. Opinion mining like other approaches, can also face some kind of challenges: First: A word can sometime represent positive opinion, and the same word in another condition may represent negative feeling. For instance, the word “Hot”. If a someone said a cup of tea is hot, that would be a positive opinion. If the he said that the weather is hot, then, that would be is a negative opinion. These differences mean that an opinion system trained to gather opinions on one type of product or product feature may not perform very well on another. Second challenge is referring to small differences between words. For example, negations. Most text mining processes, assume that small differences between two sentences don’t change meaning significantly, which is completely wrong assumption. Just look at the small example of negation in these two sentences: “I like my job” has totally different sentiment than “I don’t like my job”.

In general, an opinion is a private viewpoint representing an individual’s personal beliefs about a specific topic, item, brand, etc. (Khan, 2014). Liu et al. (2012) state that people’s opinions can exert great influence on the organizational decision-making process. The core task of opinion mining is to categories a document's text as subjective or objective (Montoyo, 2012). The core task of opinion mining is to categories a document's text as subjective or objective (Montoyo, 2012). Recently, online opinions have been analysed using sentiment analysis (SA), a type of natural language processing (NLP) that identifies the sentiment of the given text as positive, neutral, or negative (Mostafa, 2013).
SA can be influential in fields such as product review (Yi, 2003), movie review (Na, 2010), political information (Tumasjan, 2011), and stock market prediction (Bollen, 2011; Nassirtoussi, 2014). The first step in this process is to identify whether the sentence is subjective or objective. If the sentence is subjective, SA will determine whether the sentence expresses positive or negative opinions (Medhat, 2014). Three main classification levels exist in SA: document-level, sentence-level, and aspect-level. The task at document-level is to classify whether a whole opinion document expresses a positive or negative sentiment (Pang, Lee and Vaithyanathan, 2002). Sentence level (tweet-level) focuses on sentences and determines whether each sentence expressed a positive, negative or neutral opinion. Instead of looking at language constructs, aspect level (entity level) directly looks at the opinion itself (Liu, 2012). In this study, we will use the hybrid approach of Sentence level, and aspect (entity) level for sentiment analysis.

This thesis introduces a novel approach, called NormalSentiment, which captures the contextual semantics of words to calculate a confidential interval for the contextual sentiment of a specific word (for example a specific brand, such as an iPhone). For this aim, I used a precompiled and known lexicon, called SentiWordNet (Baccianella, 2010), as a comprehensive lexicon to determine the prior sentiment of words. I restricted the recommended approach to just the first SA task (entity level), which can be expanded for tweet-level.

I evaluated the recommended approach using Twitter as a comprehensive dataset and SentiWordNet as a thorough lexicon for the two most famous brands of cell phones (iPhone and Galaxy) to determine customers’ general sentiments about these brands. We then will compare the performance of the normal sentiment approach with the SentiCircle approach, which was constructed by Saif et al. (2015). The experimental result demonstrated that the proposed method perform so much better than the SentiCircle method. This means that despite the fact that SentiCircle is a very accurate and precise method the normal sentiment method is more accurate.
The final goal of this thesis, is to find answer for the following research question:

1. **How to calculate the latent sentiment of a given word, according to its dependency of the context it belongs to?**

The remainder of this thesis is structured as follows. In literature review (section 3), I will try to review the existing approaches of opinion mining and sentiment analysis, and also will review the pros and cons of each approach by finding the gaps. Section 4 will be demonstrating the theoretical framework of the thesis and proposed approach, by introducing the model and the analytics I did for that. In section 5, I will explain what was the methodology to capture the data. Section 6 is elaborating my proposed approach with details and formulas applied on data in order to reach out to the result. Section 7 is demonstrating the result and accuracy of the proposed approach by some metrics. Finally, I will conclude the whole thesis in section 7 and 8.
3. Literature Review

In this section, I will review the related works in opinion mining and sentiment analysis, which are classified as following.

3.1. Sentiment analysis in text

The techniques utilized by former authors/scientists for SA can be roughly divided to lexical approaches and learning-based approaches.

3.1.1. Machine learning approach

Although a machine learning model may apply a mix of different techniques, the methods for learning can typically be categorized as three general types:

- **Supervised learning**: The learning algorithm is given labeled data and the desired output. For example, pictures of cats labeled “cat” will help the algorithm identify the rules to classify pictures of cats.
- **Unsupervised learning**: The data given to the learning algorithm is unlabeled, and the algorithm is asked to identify patterns in the input data. For example, the recommendation system of an e-commerce website where the learning algorithm discovers similar items often bought together.
- **Reinforcement learning**: The algorithm interacts with a dynamic environment that provides feedback in terms of rewards and punishments. For example, self-driving cars being rewarded to stay on the road.

The machine learning (ML) approach relies on ML algorithms to solve the SA as a regular text classification problem that makes use of syntactic and/or linguistic features (Medhat, 2014). The two main ML approaches are the Support Vector Machine (SVM) and the Naive Bayes (NB) method. SVM constructs a hyperplane or a set of hyperplanes in an infinite dimensional space, which can be used for classification (Bhadanea, 2015). The NB method is one of the approaches that have been found to achieve maximum accuracy (Hatzivassiloglou., 2003). These two supervised ML algorithms are strong sentiment
recognition methods in text, and, of course, some studies have found that SVM outperforms NB in some tasks (Pang, 2002; Alm, 2005).

**Caveats:** However, these two methods have some weaknesses: they disregard negation, syntactic relations, and semantic dependencies. Moreover, their process is time-consuming and annotation is too difficult (Ghazi, 2014).

### 3.1.2. Lexicon-based approach

The lexicon-based method often uses predefined dictionaries of terms annotated with “positive” or “negative” scores, indicating the strength (i.e., polarity) of the sentiment they represent. This approach has two main automated approaches: a dictionary-based method and a corpus-based method. The corpus-based method is performed using a statistical approach or a semantic approach as illustrated in the subsections below.

#### 3.1.2.1. Statistical approach

The statistical approach identifies the co-occurrence pattern of the set of words using statistical techniques. Fahrni and Klenner (2008), have derived the posterior polarity of adjectives in a corpus using their co-occurrence pattern. If the word occurs more frequently among positive texts, then its polarity is positive, and if it occurs less frequently its polarity is negative (Fahrni and Klenner, 2008).

Turney (2002) established a widely accepted hypothesis that:

- *if two words frequently appear together in similar contexts, they tend to have the same polarity.*

Thus, by the co-occurrence pattern of words, the polarity of an unknown word can be extracted. This could be done by using the point-wise mutual information (PMI) method (Turney, 2002). I have used the consequences of this hypothesis in our research in order to illustrate the co-occurrence pattern of words.
3.1.2.2. Semantic approach

The semantic orientation of words is a subset of the statistical approach, using the PMI method (Turney, 2003), which can be categorized into contextual and conceptual methods. In addition, Lund and Burgess (1996) have proposed a Hyperspace Analogue to Language (HAL) that uses semantic space. In this space, words are represented by points along each axis depending on the meaning of those words (Lund; Burgess, 1996).

_Caveats: The main weakness of this approach is its inability to assign sentiment to some words with a specific orientation. For example, it can't distinguish the word, "hot", as being either a positive sentiment for a cup of tea or a negative sentiment when someone is uncomfortable at noon on a summer day._

Considering all the weaknesses of both the ML and the lexicon-based approaches, This thesis is aimed to produce a hybrid method that combines several ways of tackling the problem.

- First, the proposed method solves the problem of the independency of words and the time-consuming nature of ML.
- Next, this approach can assign the sentiment of the words with a specific domain orientation.
- Finally, it can be easily applied to different domains and diverse datasets.

3.2. Stock market prediction with SA

The welfare of all societies depends on their market economies. Therefore, it is essential to study markets and their movements. Toward this end, Arman et al. (2014) have introduced a systematic review to predict markets and their movements using text mining and sentiment analysis methods. Heeryon et al. (2014) also presented a different lexicon-based approach to classify product reviews using existing sentiment lexicons, which merges some multiple lexicons by incorporating labeled product reviews to enable domain adaptation of sentiment values. In this case, gathering information about customers’ opinions about a specific brand is crucial for a company’s decision-makers. Valeria et al. (2015) showed that the easiest and cheapest way to learn about customers’
opinions is by accessing online social media sites, such as Twitter or Facebook. In this context, Bollen, Mao, and Zeng (2011) found that the aggregation of millions of tweets posted daily on Twitter can be used to predict changes in the stock market over time.

In the cell phone industry, identifying and measuring customer satisfaction of mobile services is very popular and essential (Kang, 2014). Thus, this thesis is aimed to introduce a normal sentiment method to precisely learn about the opinions or sentiments that people using Twitter have about a specific brand, such as iPhones and Android phones (Galaxy, Nokia, LG, etc.).

3.3. Sentiment lexicons

A lexicon is a vocabulary of sentiment words with their respective sentiment polarity and strength value. The lexicon creation begins with an initial list of words, also known as seed words, and the list is extended using the synonyms and antonyms of the seed words. The synonyms and antonyms are taken from the WordNet dictionary (Ravi, 2015). The lexicons can be roughly divided into ontology based words and non-ontology based words. SA researchers have constructed numerous lexicons to classify the positive sentiments and the negative sentiments of words in the text. The most famous lexicons are: SentiWordNet (Baccianella, 2010), AFINN (Nielsen, 2011) Opinion Lexicon (M. Hu, 2004), SentiSence (Albornoz, 2012), SenticNet (Cambria, 2012), MPQA subjectivity lexicon (Wilson, 2005), and the Thelwall-Lexicon (Thelwall, 2010; 2012). In addition, Heeryon et al. (2014) merged these multiple sentiment dictionaries in order to predict markets using ‘merge’, ‘remove’, and ‘switch’ operations. Their method compares the positive/negative review’s dictionary word occurrence ratios with the positive/negative review ratio to determine which entry words should be removed and which of the sentiment polarities of the entry words should be switched (Heeryon, 2014).

In this thesis, I use the SentiWordNet lexicon in order to achieve the prior sentiment score of the terms regardless of their context.
4. Theoretical Framework

As mentioned in the prior subsections, the existing lexicons only offer a general sentiment for a supposed term (depending on the natural positivity or negativity of the term). For example, using SentiWordNet to obtain the general sentiment score of the term "amazing", resulted in an original sentiment score of +0.75. However, we know that the real sentiment of a term isn’t static; it depends 2 things:

- on the type of context (contextual semantic) (Saif, 2015).
- It also depends on the other terms that appear in the text (co-occurrence pattern) (Turney P., 2002).

This thesis is aimed to introduce a new approach, called the normal sentiment method, which considers the limitations mentioned in previous section. This method identifies the sentiment orientation of the terms according to the context in which the term has been used. I define context as the set of tweets which have been produced by customers on a social media platform (Twitter in this thesis).
4.1. Assumptions

The assumptions in this thesis are as following:

- Tweets represented as $T_k, k = 1:K$

- If two words frequently appear together in a similar context, they tend to have the same polarity. So by using the co-occurrence pattern of words, the polarity of an unknown word can be extracted (Turney, 2002; Wittgenstein, 1953).

- Term Vector $W = (w_1, w_2, ..., w_n)$ is defined as the vector of terms that occur with our supposed term "V", in the tweet "$T_k$". where $w_i$ is a symbol to indicate the most frequently used terms that co-occur with term "V".

Saif et al. (2015) found a relationship between the centre term "V" and the term $w_i$. In the proposed method in this thesis, we need these relationships to illustrate the NormalSentiment. In order to do so, we must first define that formula and its precedents. For this aim, we need two main features for $w_i$:

- Prior sentiment score ($P(w_i)$): These scores are available in the lexicons that we introduced in sub-section 3.3. In this thesis, I use SentiWordNet as a reference lexicon.

- Degree of co-occurrence ($C$): The degree of correlation between "V" and "$w_i$" can be computed as followings (Saif, 2015):

$$C(V, w_i) = L(V, w_i)_k \cdot \log \frac{N}{N_{wi}}$$
Where $L(V, w_i)_k$ is the number of times which the term "V" occurs with "w_i" in the $T_k$, $N$ is the total number of terms in the sampling, and $N_{wi}$ is the total number of terms that occur with "w_i".

4.2. NormalSentiment approach

4.2.1. The position of words in the polar coordinate systems

In this section, I want to find the position of the terms wi in the Cartesian coordinate system. For this aim, the polar system is more consistent when using the two main features of "wi" ($P(wi), C(V, wi)$) which was explained in section 4.1. Then, the polar system must move to Cartesian system because the normal distribution graph can be illustrated only on the Cartesian coordinate system. The NormalSentiment method wants to compute the contextual semantic sentiment for the target word "V". As a result, this term is situated at the center of the coordinates (0,0). I define the two components of the polar system as:

\[ R_i: C(V, wi) \]  \hspace{1cm} (2)

\[ \theta_i: P(wi) \frac{\pi}{2} \]  \hspace{1cm} (3)

where $R_i$ is the length of the vector that will be plotted from the centre of the coordinate to the position of the term "V", and $\theta_i$ is the angle between this vector and the vertical axis. The formula of $R_i$ calculation has been designed so that the length of the vector restricts to a scale between 0 and 1; moreover, the sentiment scores of the SentiWordNet, which we use in this thesis, are limited to a scale between 0 and 1. Therefore, $\theta_i$ will be always limited to a scale between $(-\pi/2, +\pi/2)$.

All of the terms will be positioned in the first and second quadrants, based on their prior sentiment score ($P(wi)$). The positions of the terms "V" and "wi" are shown in Fig. 1.
4.2.2. Axis labelling in two sentiment regions

Fig. 1 shows the two regions to the left and right of the term "V". We can call these two regions "sentiment regions". The right quadrant represents positive sentiment, and the left quadrant represents negative sentiment. According to equations (2) and (3), I can call the horizontal axis the sentiment axis and the vertical axis the sentistrength axis. Now I can move from the polar coordinate system to the Cartesian system using equations (4) and (5) below:

\[
\begin{align*}
SS(w_i) &= R_i \cdot \cos(\theta_i) : \text{Sentiment strength (Vertical axis)} \\
S(w_i) &= R_i \cdot \sin(\theta_i) : \text{Sentiment of the term (horizontal axis)}
\end{align*}
\]

It is obvious that, if a supposed term "wi" lies in the right quadrant, it will represent positive sentiment. In addition, a greater distance from the center indicates a higher positive sentiment score. If "wi" lies in the left quadrant, it represents negative sentiment with greater distance from the center indicating a higher negative sentiment score. That is because, if \(\theta_i\) increases, \(\sin(\theta_i)\) increases mutually, so it will gain a higher sentiment score (equation 5).
This thesis can claim that the terms "wi" are distributed on the two sentiment regions in a way that they make normal distribution. The following paragraph proves this claim.

As mentioned in sub-section 4.1, if two words frequently appear together in similar contexts, they tend to have the same polarity (Turney, 2002). As such, we can define "Swi" (sentiment of the term "wi", or horizontal component of this term on the sentiment region) as a sequence of some independent random variables, all of which have equal expectation ($\mu$) and equal variance($\sigma$). Moreover, I define "SV" as the total sentiment of the context (set of tweets) related to the word "V". So, "SV" is computed as the aggregation of the sentiment of each term in the context. Then we have the following equation:

$$SV = \sum Swi = Sw1 + \ldots + Swn$$  \hspace{1cm} (6)

According to the central limit theorem, if the number of the components of the term vector "n", is bigger than 30, the distribution of "SV" will be normal with ($\mu, \sigma$) as its components (Fig. 2). As a result, the distribution of the terms that lie on the sentiment regions is normal with the components of ($\mu, \sigma$).

\textit{Figure2. NormalSentiment Distribution for each term (v) (Own Reference)}
Normal distribution properties for sentiment analysis

- Normal distribution function

I can use equation (7) to compute percentage of terms "$w_i$" which their sentiments limit to a particular scale.

\[
f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/(2\sigma^2)}, \quad x \in R
\]  

(7)

Where "$X$" is considered here as "$SV$", and $(\mu, \sigma)$ are the expectation and standard variation of the normal distribution.

For example, Fig. 3 shows a hypothetical distribution for the term "$V$". Now we can compute the percentage of the terms "$wi$", which have been selected from the sample tweets (context), and their sentiments limit to a scale between $(-0.06, -0.02)$.

![Figure 3. Integrated surface between 2 sentiment scores (Own Reference)](image)

- Application of the area under normal curve

The high percentage of the terms that lie between two positive sentiments in a hypothetical normal distribution shows that most of the users (customers) feel positive about the term "$V$" (as previously mentioned, "$V$" can be a particular brand). In other words, the dominant sentiment of the term, according to the type of the context and its
co-occurrence degree, is positive. For instance, if the term "V" is "iPhone" and the percentage of the terms that lie between two positive sentiments is 90%, the users (who are a sample of customers) express generally positive feelings, and they are satisfied with this brand. Inversely, if the percentage of the terms that lie between two negative sentiments were high, it would show a generally negative sentiment of this term and customers’ dissatisfaction with this brand.

➢ **Expectation of normal distribution**

Consider a hypothetical term vector which is used to plot sentiment distribution. The number of the terms "wi" in this vector is "n". The expectation of such a normal distribution will be computed by using the following formula:

\[ \mu = \frac{\sum_{i=0}^{n} SW_i}{n} \]  

(8)

The expectation (\(\mu\)), as a feature of normal distribution, shows the average sentiment of tweeter users about the term "V". The positivity of expectation shows that the average of existing sentiments "SW_i" about the term "V" is positive. Also, a higher amount of \(\mu\) shows the better general sentiments of tweeter users about this term (Fig. 2).

Conversely, if \(\mu \leq 0\), then we can understand that the average sentiment of tweeters about this term is negative, indicating that tweeters have a bad attitude about this term (Fig. 3).

➢ **Standard variation of normal distribution**

The following formula is used in statistical science to compute the standard variation of distributions:

\[ \sigma = \int_{-\infty}^{\infty} (SW_i - \mu)^2 f(SW) dSW \]  

(9)
This feature of normal distribution shows scattering of all sentiments $S_{wi}$, to their average. A smaller value of $\sigma$ shows that the distribution is concentrated on the average. As an example, if $\mu$ for the term "V": iPhone" is computed as 0.06 with a small standard variation, then the sentiment of all tweeter users about this brand is approximately 0.06. That is, most of the customers are unanimous in their sentiment about the iPhone brand (Fig. 4).

![Concentrated distribution](image1)

*(Own Reference)*

Also the high value of $\sigma$ shows that the sentiment of people about the term "V" is so distributed and their sentiment is not unanimous on that. In fact, in such a distribution, the sentiment of people about this term, depending on their preferences is so variated, some of them have so positive and good sentiment and some have negative and bad sentiment. Fig.5.

![Sporadic distribution](image2)

*(Own Reference)*
5. Methodology

Twitter is a microblogging service that was launched on July 13, 2006. The maximum size of each microblog is 140 characters, approximately the length of a headline. According to, (Semiocast.com (2012)), a marketing research company, announced there are now around 500 million active Twitter users. Thelwall, et al. (2011) found that more than 80% of users update their followers on either what they are actually doing at that time or disseminate information regarding daily experiences. Since Twitter is the largest, most popular and well-known microblogging site, it was selected to conduct the analysis reported in this study.

My tweet sampling method was to first collect a set of tweets that comprised the term "V". According to the assumption in sub-section 4.1, I have provided my context. In this context, we determined the most frequently used terms. Therefore, the Term-Vector can be constituted.

In this thesis, I aimed on 2 terms with [#iPhone & #Galaxy].

The data used in this thesis, is collected from twitter. First I have created a twitter application in apps.twitter.com website. After filling all the details and verification, I was granted access to the customer’s opinion/tweets and access keys including: Consumer key (API key), Access token, and owner ID.

Based on findings by Huang, Thornton, and Efthimiadis (2010) that tweeters invented the hashtag in 2008 to help them find a particular subject or specific tweet (Huang, 2010), Now I can use my credential keys in R language to extract customers opinion data. In order to do so, I will use (#iPhone & #galaxy) as described in following section.

In the mentioned time interval, application acquired 1,000,000 tweets for the term "iPhone" and 1,000,000 tweets for the term "Galaxy". To guarantee the randomness of our data, the tweet selection was varied by day of the week and time of day. After gathering the data, I selected 1000 tweets for each term, (totally 2000 tweets) to form a comprehensive dataset to implement my proposed approach. The sample size in this
thesis is comparable to the dataset used by Qiu (2010), which included 3783 opinion sentences, or the research of Mostafa (2013), which comprised 3516 tweets without any filtering or recollecting. Table 1 shows the size of collected dataset. In the following section, I generally survey the dataset which was prepared.

<table>
<thead>
<tr>
<th></th>
<th>Sample tweets</th>
<th>random tweets</th>
</tr>
</thead>
<tbody>
<tr>
<td>#iPhone</td>
<td>1,000,000</td>
<td>1000</td>
</tr>
<tr>
<td>#Galaxy</td>
<td>1,000,000</td>
<td>1000</td>
</tr>
</tbody>
</table>

Table 1. Tweet Sampling

6. Elaboration of the central topic

In this section, analysis is applied in 2 different methods:

- First is the common data annotation, using R Package, by calculating Sentiment scores and plotting in Sentiment Regions. (This is the common approach which is already done by other, or in the other word, it is not contextual semantics approach)

- Second I can apply NormalSentiment method, using AlchemyAPI.com website form IBM, which includes sentiment scores and relevance of each keyword.
6.1. First Approach: Data Annotation with R package:

It has to be emphasized this approach is calculating sentiment score independently with the other words in the context. So it cannot be considered as contextual semantics approach.

Dataset collection method:

In order to extract data of Twitter we need to create a Twitter Application (https://developer.twitter.com/en/apps). After creating application and getting API Access Tokens, we are ready to implement in R package.

For this purpose, the following packages are required:

- ROAuth: Provides an interface to the OAuth 1.0 specification, allowing users to authenticate via OAuth to the server of their choice.

- Twitter: Provides an interface to the Twitter web API.

Now it is ready to implement tweet collection, pre-processing, and analysis using R:

```r
# Required packages:

install.packages("twitteR")
install.packages("ROAuth")
library("NLP", lib.loc="~/R/win-library/3.3")
library("twitteR", lib.loc="~/R/win-library/3.3")
library("syuzhet", lib.loc="~/R/win-library/3.3")
library("tm", lib.loc="~/R/win-library/3.3")
library("SnowballC", lib.loc="~/R/win-library/3.3")
library("stringi", lib.loc="~/R/win-library/3.3")
library("topicmodels")
library("syuzhet", lib.loc="~/R/win-library/3.3")library("twitteR")
library("ROAuth")
```
Now the following command will extract n tweets, with my require hashtags (#iPhone & #Galaxy) and we call dataset as tweets_iPhone:

#Tweet Extraction:

tweets_iPhone <- searchTwitter("#iPhone", n=1000, lang = "en")
tweets_Galaxy <- searchTwitter("#Galaxy", n=1000, lang = "en")

The problem is that tweet_iPhone is not a dataframe and and cannot be manipulated easily. Then it needs to be converted to dataframe with the following line:

#Convert to Dataframe

iPhone_tweets <- twListToDF(tweets_iPhone)
Galaxy_tweets <- twListToDF(tweets_Galaxy)

The acquired dataframe, has about 20 columns, with too many links, mentions, stop words, punctuations, upper/lower letters, and blanks. In the R language, tm package functions, cannot be executed with these kind of data. In order to make it executable by tm functions, we need to apply the following Pre-processing steps: (The same should be executed for Galaxy dataset):

#Pre-processing:

iPhone_text<- iPhone_tweets$text
In order to remove stop words, here we start applying tm package which is going to be used for text mining and sentiment analysis for the rest of the lines. Tm_map is the function to be used in this part:

```r
# convert all text to lower case
iPhone_text <- tolower(iPhone_text)

# Replace blank space ("rt")
iPhone_e_text <- gsub("rt", "", iPhone_text)

# Replace @UserName
iPhone_text <- gsub("\@\w=*", "", iPhone_text)

# Remove punctuation
iPhone_text <- gsub("[[:punct:]]", "", iPhone_text)

# Remove links
iPhone_text <- gsub("http\:\w+=", "", iPhone_text)

# Remove tabs
iPhone_text <- gsub("[ \t]{2,}", "", iPhone_text)

# Remove blank spaces at the beginning
iPhone_text <- gsub("^ ", "", iPhone_text)

# Remove blank spaces at the end
iPhone_text <- gsub(" $", "", iPhone_text)
```

In order to remove stop words, here we start applying tm package which is going to be used for text mining and sentiment analysis for the rest of the lines. Tm_map is the function to be used in this part:

```r
# clean up by removing stop words
iPhone_tweets.text.corpus <- tm_map(iPhone_tweets.text.corpus, function(x) removeWords(x, stopwords()))
```
Now we finished with pre-processing steps, and ready to do the Sentiment Analysis. In order to do so, `get_nrc_sentiment` function is required which can be found in `tm` library. The sentiments will be stored in the variable which is called `mysentiment_iPhone`:

```r
# getting sentiment/emotions
mysentiment_iPhone<-get_nrc_sentiment(iPhone_text)
```

Next lines, is calculating total scores of sentiments to be stored in `Sentimentscores_iPhone` variable:

```r
# Total Score calculation for Sentiments
Sentimentscores_iPhone<-data.frame(colSums(mysentiment_iPhone[,]))
```

```r
names(Sentimentscores_iPhone)<-"Score"
Sentimentscores_iPhone<-cbind("sentiment"=rownames(Sentimentscores_iPhone),Sentimentscores_iPhone)
rownames(Sentimentscores_iPhone)<-NULL
```

Now we are done with Pre-processing and gettion total scores of terms, and ready to plot the acquired sentiment scores in Sentimental Regionns. The visualization commands are in the following lines:

```r
# plotting the sentiments regions
ggplot(data=Sentimentscores_iPhone,aes(x=sentiment,y=Score))+geom_bar(aes(fill=sentiment),stat = "identity")+
  theme(legend.position="none")+
  xlab("Sentiments")+ylab("scores")+ggtitle("Sentiments of people behind the tweets on tech giant iPhone")
```

The output of the analysis can be shown in the sentiment regions Figure6.
Figure 6 shows iPhone user’s sentiment regions annotated with their feeling about the product. So from this figure, we can understand iPhone has a huge amount of positive feedbacks from customers. But also there are still too much people who have negative feeling and not satisfied with the product. These negative opinions are about less than half of the positive feedbacks. Another useful interpretation is the people who lsust on this brand are aslo significant. It means we have trust feedbacks as much as negative and dissatisfaction feedbacks.

Figure 6. #iPhone Sentiment Regions (R package output and self owned)
Figure 7 demonstrates Galaxy users sentiment regions. It shows more than 70% of the feedbacks where positive, and we can conclude that Galaxy had a huge amount of positive impact among customers. Also the negative opinions are too much less that positive feedbacks which can be ignored (they might be customers who have loyalty to other brands like iPhone, Nokia,… who are making negative feedbacks. The amount of trust feedbacks among Galaxy customers is considerable as well.

Figure 7. #Galaxy Sentiment Regions (R package output and Own reference)
Comparing 2 brands feedbacks (6.1. conclusion):

Using the above charts, I can perform Sentiment analysis, regarding customer’s opinion of iPhone and Galaxy. As shown above, iPhone users have positive opinion with 200 score, and negative with 100 score. It means that positive opinion is 2 times more that negative opinions.

But for Galaxy customers, positive opinion is 1200 and negative is only 200.

It shows that Galaxy users are much more satisfied than iPhone users in application. Galaxy users have much more positive feedbacks on the brand they are using and their trust feedbacks are also more than iPhone users.

In this situation, iPhone stakeholders should figure out why the proportion of positive to negative feedbacks is 2:1, and what can be the root cause of this amount of dissatisfaction.
6.2. Second Approach: NormalSentiment Approach (Contextual Semantics):

As mentioned in the last section, the real sentiment of the term "V" is not static at all. Its value depends on the context in which it was previously used. In this section, we want to plot the NormalSentiment graph for two terms: "iPhone" and "Galaxy".

6.2.1. Using a machine learning approach:

I used "AlchemyAPI.com" which is an online service that allows for the extraction of most frequent entities from a text along with their associated semantic concept class. This service uses some supervised machine learning algorithms to identify the most frequent semantic concept. The algorithm is supervised because it is using SentiwordNet lexicon, as a prior sentiment score of terms and can be considered as labels. I executed our required dataset in this service to identify 3 factors:

- the most frequent entities in the text,
- their relevance percent,
- and the real sentiment of each entity.

Some of iPhone and Galaxy dataset output from AlchemyAPI are shown below.
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Relevance</th>
<th>Sentiment</th>
</tr>
</thead>
<tbody>
<tr>
<td>View photo</td>
<td>0.9622771</td>
<td>positive</td>
</tr>
<tr>
<td>iphone</td>
<td>0.823728</td>
<td>mixed</td>
</tr>
<tr>
<td>favorites</td>
<td>0.777552</td>
<td>mixed</td>
</tr>
<tr>
<td>retweets</td>
<td>0.672278</td>
<td>mixed</td>
</tr>
<tr>
<td>new iPhone</td>
<td>0.580732</td>
<td>positive</td>
</tr>
<tr>
<td>image permalink</td>
<td>0.568916</td>
<td>positive</td>
</tr>
<tr>
<td>iphone giveaways</td>
<td>0.540429</td>
<td>neutral</td>
</tr>
<tr>
<td>iphone giveaway</td>
<td>0.525885</td>
<td>positive</td>
</tr>
<tr>
<td>iphone ort</td>
<td>0.523603</td>
<td>positive</td>
</tr>
<tr>
<td>custom pink iphone</td>
<td>0.516585</td>
<td>positive</td>
</tr>
<tr>
<td>iphone chargers</td>
<td>0.512543</td>
<td>negative</td>
</tr>
<tr>
<td>iphone charger</td>
<td>0.511327</td>
<td>negative</td>
</tr>
<tr>
<td>BLACK IPHONE</td>
<td>0.508272</td>
<td>positive</td>
</tr>
<tr>
<td>white iphone</td>
<td>0.501354</td>
<td>mixed</td>
</tr>
<tr>
<td>iphone news</td>
<td>0.500974</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone 6s</td>
<td>0.497517</td>
<td>mixed</td>
</tr>
<tr>
<td>best iPhone news</td>
<td>0.494799</td>
<td>positive</td>
</tr>
<tr>
<td>10th iPhone charger</td>
<td>0.490965</td>
<td>neutral</td>
</tr>
<tr>
<td>raspberry white iphone</td>
<td>0.484273</td>
<td>negative</td>
</tr>
<tr>
<td>Actualidad iPhone</td>
<td>0.483704</td>
<td>neutral</td>
</tr>
<tr>
<td>iPhone Tricks</td>
<td>0.480776</td>
<td>neutral</td>
</tr>
<tr>
<td>iPhone weather ago</td>
<td>0.477906</td>
<td>negative</td>
</tr>
</tbody>
</table>

Figure 8: iPhone dataset Output (AlchemyAPI output and own reference)
<table>
<thead>
<tr>
<th>iPhone Charger</th>
<th>0.475918</th>
<th>negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check iPhone Battery</td>
<td>0.473763</td>
<td>neutral</td>
</tr>
<tr>
<td>PINK iPhone 5s</td>
<td>0.469922</td>
<td>positive</td>
</tr>
<tr>
<td>KRY iPhone 5/5s</td>
<td>0.469382</td>
<td>negative</td>
</tr>
<tr>
<td>iPhone FAV</td>
<td>0.468271</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone burn</td>
<td>0.467392</td>
<td>positive</td>
</tr>
<tr>
<td>undressed iphone</td>
<td>0.467195</td>
<td>negative</td>
</tr>
<tr>
<td>iPhone update</td>
<td>0.466415</td>
<td>negative</td>
</tr>
<tr>
<td>new iPhone lightning</td>
<td>0.445825</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone 4S</td>
<td>0.465824</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone case</td>
<td>0.464838</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone new emojis</td>
<td>0.464707</td>
<td>negative</td>
</tr>
<tr>
<td>bidee iPhone</td>
<td>0.463736</td>
<td>negative</td>
</tr>
<tr>
<td>iPhone Cases</td>
<td>0.4628</td>
<td>negative</td>
</tr>
<tr>
<td>iPhone flashlight</td>
<td>0.462264</td>
<td>negative</td>
</tr>
<tr>
<td>moms iPhone</td>
<td>0.461769</td>
<td>positive</td>
</tr>
<tr>
<td>latest iPhone</td>
<td>0.460373</td>
<td>negative</td>
</tr>
<tr>
<td>iPhone News Daily</td>
<td>0.459633</td>
<td>neutral</td>
</tr>
<tr>
<td>iPhone owners</td>
<td>0.436882</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone 420s</td>
<td>0.458078</td>
<td>negative</td>
</tr>
<tr>
<td>naked iPhone</td>
<td>0.458514</td>
<td>neutral</td>
</tr>
<tr>
<td>iphone quality stinks</td>
<td>0.458302</td>
<td>negative</td>
</tr>
<tr>
<td>Allegedly Preparing iPhone</td>
<td>0.457967</td>
<td>negative</td>
</tr>
<tr>
<td>Team_Kaliber iPhone Background</td>
<td>0.45729</td>
<td>positive</td>
</tr>
<tr>
<td>iPhone emojis</td>
<td>0.456878</td>
<td>negative</td>
</tr>
<tr>
<td>slick new iPhone</td>
<td>0.436437</td>
<td>positive</td>
</tr>
<tr>
<td>iphone recordings</td>
<td>0.456028</td>
<td>negative</td>
</tr>
<tr>
<td>Phone Hacks</td>
<td>0.433574</td>
<td>negative</td>
</tr>
</tbody>
</table>

Figure 8: iPhone dataset Output (AlchemyAPI output and own reference)
<table>
<thead>
<tr>
<th>Term</th>
<th>Score</th>
<th>Sentiment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOG Galaxy support</td>
<td>0.384831</td>
<td>neutral</td>
</tr>
<tr>
<td>ur Galaxy pants</td>
<td>0.383787</td>
<td>positive</td>
</tr>
<tr>
<td>faint galaxy structure</td>
<td>0.381619</td>
<td>negative</td>
</tr>
<tr>
<td>galaxy painting</td>
<td>0.381356</td>
<td>negative</td>
</tr>
<tr>
<td>Galaxy Macau</td>
<td>0.378838</td>
<td>neutral</td>
</tr>
<tr>
<td>cheaper Galaxy Gear</td>
<td>0.377853</td>
<td>positive</td>
</tr>
<tr>
<td>edition Iron Man</td>
<td>0.37765</td>
<td>positive</td>
</tr>
<tr>
<td>Way Galaxy Shimmers</td>
<td>0.377237</td>
<td>positive</td>
</tr>
<tr>
<td>beautiful galaxy</td>
<td>0.376993</td>
<td>positive</td>
</tr>
<tr>
<td>limited edition iron</td>
<td>0.375799</td>
<td>neutral</td>
</tr>
<tr>
<td>LA Galaxy pay</td>
<td>0.375106</td>
<td>positive</td>
</tr>
<tr>
<td>GALAXY HANKER</td>
<td>0.374382</td>
<td>negative</td>
</tr>
<tr>
<td>Galaxy Lacrosse Co.</td>
<td>0.373815</td>
<td>neutral</td>
</tr>
<tr>
<td>galaxy bedding</td>
<td>0.373057</td>
<td>positive</td>
</tr>
<tr>
<td>Cancer Galaxy Girl</td>
<td>0.372523</td>
<td>negative</td>
</tr>
<tr>
<td>Galaxy Hanging Cver</td>
<td>0.372241</td>
<td>neutral</td>
</tr>
<tr>
<td>polar ring galaxy</td>
<td>0.371799</td>
<td>neutral</td>
</tr>
<tr>
<td>Galaxy <a href="http://biz/us/1Kjs0r">http://biz/us/1Kjs0r</a></td>
<td>0.371042</td>
<td>positive</td>
</tr>
<tr>
<td>Galaxy Tree</td>
<td>0.370334</td>
<td>positive</td>
</tr>
<tr>
<td>Galaxy S5</td>
<td>0.369451</td>
<td>negative</td>
</tr>
<tr>
<td>luminous galaxy</td>
<td>0.369213</td>
<td>positive</td>
</tr>
<tr>
<td>Galaxy note</td>
<td>0.368997</td>
<td>negative</td>
</tr>
<tr>
<td>Galaxy Custom</td>
<td>0.368886</td>
<td>negative</td>
</tr>
<tr>
<td>Image gallery thumbnail</td>
<td>0.368101</td>
<td>neutral</td>
</tr>
</tbody>
</table>

Figure 9: Galaxy dataset Output (AlchemyAPI output and Own reference)
6.2.2. **Data annotation:**

Using the generated set of the most frequent terms, I can construct two term vectors for the two centre terms: iPhone and Galaxy. “wi” represent the output of the AlchemyAPI algorithm. Because not all of the extracted entities from AlchemyAPI existed in the lexicon I used (SentiWordNet), I placed only existing entities in the graphs. In addition, Table 4 shows the real subjectivity and polarity of all entities which have been extracted by AlchemyAPI.

Then I compute manually the co-occurrence degree of two terms "V" and "wi" by formula (1), using SentiWordNet as a reference lexicon. Then I used formulas 4 and 5 to identify the placement of terms on the graph.

Finally, I plotted two NormalSentiment distributions for the terms "iPhone" and "Galaxy". Fig. 10 and Fig. 11 show the NormalSentiment distributions for the iPhone dataset and the Galaxy dataset.
6.2.2.1. NormalSentiment for iPhone

Bellow chart, is the NormalSentiment distribution for iPhone dataset which has been visualized by Minitab. The following results can be concluded:

- The total Sentiment score of iPhone users lies between -1.5 to 2. Approximately half of the population have negative feedbacks and the rest have positive feedbacks.
- The Average of iPhone users Sentiment score, is 0.2. It is too much near to zero. We can conclude that the average sentiment of the iPhone users is Neutral. Or in the other words, most of the iPhone users have neutral opinion about iPhone. But also the positive feedbacks are stronger than negative ones. Because the average is still positive.
- Standard Variation of the population is 0.7. It is such a high variation and means that iPhone users are not unanimous in their opinion regarding iPhone. Some of them has 2 and some of them has -1.5. In the other words, some of them has a very positive opinion and like the brand so much, and some of them really hate the brand and representing strong negative opinion.

*Figure 10. NormalSentiment for iPhone (Minitab output and own reference)*
6.2.2.2. NormalSentiment for Galaxy

Bellow chart, is the NormalSentiment distribution for Galaxy dataset which has been visualized by Minitab. The following results can be concluded:

- The whole feedbacks and sentiment scores from Galaxy users, lies between -0.8 to 1.2. It is showing clearly that most of the population are in positive region. Because the max positive sentiment score, is more than max (or strongest) negative sentiment scores.
- The Average of Galaxy users sentiment scores is about zero. Clearly shows that average of the sentiments regarding Galaxy is neutral, or in the other words, most of the users have neutral sentiment regarding Galaxy brand.
- Standard Variation of the opinions score, is 0.39 which is less than iPhone distribution. It means that Galaxy users opinion are much more united than iPhone users. We cannot say that some users strongly love or hate the brand. The strongest positive score is 1.2, and the strongest negative score is -0.8.

*Figure 11. NormalSentiment for Galaxy (Minitab output and own reference)*
**Normality Test for each brand:**

In this section, kolmogorov smirnov test is used by minitab to prove the normality of distribution of each dataset.

Fig.12 shows the normality plot for iPhone, which P-Value is about 34%. If we consider the common Type1 error as 5%, then we can strongly conclude that distribution for iPhone dataset is Normal. Because P-Value is much more than Type1 Error (34% > 5%).

*Figure 12. Normality test for iPhone (Minitab output and Own reference)*
The same test and analysis is done for Galaxy Dataset.

Fig.13 shows the normality plot for iPhone, which P-Value is about 16%. If we consider the common Type1 error as 5%, then we can strongly conclude that distribution for Galaxy dataset is Normal. Because P-Value is much more than Type1 Error (16% > 5%).

Figure 13. Normality test for Galaxy (Minitab output and Own reference)
6.2.2.3. Confidence interval for expectation (Average):

In this sub-section, I want to identify a 95 percent confidence interval for the expectation of the two NormalSentiment graphs which I plotted in the earlier sub-section. For this aim, I use formulas 9 and 10 which are two well-known formulas in statistical sciences.

\[
Upper \ bound = \mu + t_{a/2,n-1} \frac{s}{\sqrt{n}} \quad (9)
\]

\[
Lower \ bound = \mu - t_{a/2,n-1} \frac{s}{\sqrt{n}} \quad (10)
\]

Where "\( \mu \)" is the expectation of distribution, "\( S \)" is the standard variation of sentiment of the samples from their average, "\( n \)" is the number of terms "\( w_i \)" in the term vector, and "\( \alpha \)" is the percentage of confidence, which we set at 0.05 in this study, and in fact it shows the first kind of error in statistical science.

Table 3 shows the upper (U) and lower (L) bounds of the confidence interval as well as the other features of the distributions.

<table>
<thead>
<tr>
<th></th>
<th>( \mu )</th>
<th>( S )</th>
<th>( N )</th>
<th>( U )</th>
<th>( L )</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone</td>
<td>0.255</td>
<td>0.74</td>
<td>51</td>
<td>0.76</td>
<td>-0.125</td>
</tr>
<tr>
<td>Galaxy</td>
<td>0.05</td>
<td>0.39</td>
<td>63</td>
<td>0.227</td>
<td>0.127</td>
</tr>
</tbody>
</table>
**Comparison of expectations of two NormalSentiment graph:**

As shown in Table 3, we computed the confidential interval for the expectation of sentiment of the terms "iPhone" as: [-0.125, 0.76], and Galaxy as: [0.127, 0.227].

The higher upper limit (U) for iPhone shows the high positive sentiment expressed by some of the customers about this brand. However, the lower limit is negative, indicating that some customers are unsatisfied.

In turn, the average sentiment of customers about the Galaxy brand is positive, but lower than the average sentiment about the iPhone. In general, we can conclude that customers expressed various sentiments about the iPhone, but their sentiments were more positive than those expressed about the Galaxy.

**Comparison of standard variations of two NormalSentiment graphs:**

It is obvious from Fig. 7 and Fig. 8 that the sentiment axis of the iPhone graph varies to a greater degree than the Galaxy graph. This variation shows that people hold a wide range of opinions about the iPhone. In contrast, the Galaxy graph is more concentrated with people holding more analogous opinions. Also Table 3 presents the quantitative support for this conclusion. It presents the standard variation for iPhone (0.74) and for Galaxy (0.39). The standard variation for the iPhone was higher than for Galaxy, indicating that customers held a greater variety of opinions about the iPhone than the Galaxy.
7. Analysis and Results

7.1. Baselines:

As described in the last sections, I extracted the prior and static sentiment score of terms from a well-known lexicon which is called SentiWordNet. But also the other researches in this field, have used other lexicons like:

- MPQA subjectivity lexicon,
- Thelwall-lexicon

Therefore, I want to investigate the ability of NormalSentiment in comparison with other methods and lexicons in other researches as well.

Then I will compare the performance of NormalSentiment method in entity level Sentiment Analysis. But first I need to clarify on 2 baseline in this area:

- Lexicon labelling Method: this method is the combination of MPQA and SentiWordNet lexicons in order to extract the prior sentiment score of terms. (contextual free). The method is working in a way that if a predefined tweet has more positive word than negative(using contextual free lexicons), it is labeled as positive, and if negative words are more, we label it as negative. So in entity level, we assign the sentiment of a term according to the number of positive and negative words which are co-occurring with the term, in a defined tweet. In this case, the method which uses MPQA lexicon is called MPQA method, and the method which uses SentiWordNet is referred to SentiWordNet method.

- SentiStrength: We can call this approach as state-of-the-art and it is lexicon based to extract the sentiment score. In this approach, each tweets have 2 sentiment strength: first sentiment is negative which is between -1 and -5. -1 can be interpreted as not very negative, and -5 is too extreme negative in sentiment score. Correspondingly, the positive strength starts from +1 which is not so much positive, to +5 which is extremely positive.
In this approach, a tweet is considered positive if the positive sentiments are more than 1.5 times higher than negative sentiment, in if not, it will be considered as negative.

Also for entity (term) sentiment extraction, the sentiment will be assigned according to the sum of positive and negative terms which occur together with the supposed term.

We should also take this fact into consideration that SentiStrength needs some pre-defined lexical rules such as some emoticons, negations, boost words (e.g., extremely, absolutely,…), and intensifiers.

The performance and accuracy of all these methods, and different application of lexicons, can be accessed in the article by Saif, H., et al (2015). Therefore, I will try to calculate the performance measurements of NormalSentiment, and compare it to other methods and approaches provided by other scientist in Sentiment Analysis area.
7.2. Experimental Results

In this thesis, I report the performance of NormalSentiment method in comparison with baselines and approaches which described in the previous sub-section.

Also this thesis is using entity level approach and SentiWordNet. This approach is somehow close to Senti-Circle (Senti-Median Method) proposed by Saif, H., et al in 2015 in term of functionality.

Senti-circle (Senti-Median) method is using SentiWordNet, MPQA, and Thelwall lexicons to identify the overall sentiment of a given term. We also can reach out to the performance of each implementation and compare it to the NormalSentiment method.

7.2.1. NormalSentiment Method Performance

In this section, I report the performance of the proposed approach, the NormalSentiment method, to identify a confidential interval for the sentiment score of the term "V". For this aim, I plotted the NormalSentiment distribution for all of the sample terms in the output of AlchemyAPI algorithm (Figure 8 and Figure9). The table of those sample sizes are shown separately to acquire the average of their sentiment score.

Table 4 shows the results of this experiment and the real sentiments of terms (obtained by using the algorithm in AlchemyAPI). The same result and performance can be reached out in Rapidminer software.

<table>
<thead>
<tr>
<th></th>
<th>positive</th>
<th>negative</th>
<th>Mixed</th>
<th>Neutral</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real sentiment</td>
<td>57</td>
<td>60</td>
<td>12</td>
<td>21</td>
<td>150</td>
</tr>
<tr>
<td>Existing in lexicon</td>
<td>45</td>
<td>54</td>
<td>6</td>
<td>15</td>
<td>120</td>
</tr>
<tr>
<td>Normal sentiment</td>
<td>69</td>
<td>51</td>
<td>-</td>
<td>-</td>
<td>120</td>
</tr>
</tbody>
</table>

Table4: The Real Subjectivity and NormalSentiment of the terms in iPhone dataset
Table 4 shows the real subjectivity and the calculated sentiment of the NormalSentiment approach in iPhone dataset.

As mentioned in previous sections, AlchemyAPI extracted 150 terms in the vector of terms, 120 words are existed in SentiWordNet lexicon. Therefore, I will focus on this 120 words.

The output of AlchemyAPI in Figure 8, shows:

- Positives: 57 positive words, which 45 is in SentiWordNet,
- Negative: 60 negative words, which 54 exist in SentiWordNet,
- Mixed: 12 mixed which 6 exists in SentiWordNet,
- Neutral: 21 neutral which 15 exists in SentiWordNet.

But also my proposed approach _NormalSentiment_ only calculates contextual semantic of sentiment scores as positive and negative.

The output of NormalSentiment has never been mixed or neutral.

As mentioned in section 6.2.2 (data annotation) how to labelling and annotation part of NormalSentiment, we will reach out to 69 positive word and 51 negative accordingly.

I used Table 4 to acquire "true positive (TP), false positive (FP), false negative (FN)", and then I computed precision and recall for our proposed NormalSentiment method using formulas 11, 12, and 13.

- True Positive: (TP):
  - Number of terms which were Positive in reality, and Normal Sentiment is calculated as positive: 57
  - Number of terms which were Negative in reality, and NormalSentiment is calculated as Negative: 51

- False Positive (FP): Number of terms which were Positive in reality, but NormalSentiment couldn’t identify clearly: 12

- False Negative (FN): Number of terms which were Negative in reality, but NormalSentiment couldn’t identify clearly: 9
Now we are ready to calculate Precision, Recall, and F-Measure according to the above mentioned numbers and following formulas:

\[
\text{Precision} = \frac{TP}{TP + FP} = \frac{57 + 51}{57 + 51 + 12}
\]  

(11)

\[
\text{Recall} = \frac{TP}{TP + FN} = \frac{57 + 51}{57 + 51 + 9}
\]  

(12)

\[
F - \text{measure} = \frac{\text{precision} \cdot \text{recall}}{\text{Precision} + \text{recall}} = 2
\]  

(13)

Now it’s the time to show the performance of NormalSentiment in Table 5 below:

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal sentiment</td>
<td>90%</td>
<td>92.3%</td>
<td>91.1%</td>
</tr>
</tbody>
</table>

*Table 5: NormalSentiment Performance results for Sentiment Analysis*
Therefore, the proposed method in this thesis (NormalSentiment) has been demonstrated the great performance measurements of 90% Precision, 92.3% of Recall, and finally 91.1% of F-Measure which are unbelievably high.

But also that will be a nice exercise if we compare this result with other the results obtained by other approaches from other scientists and researchers. So the following subsection will do the comparison between the methods.

### 7.2.2. Comparison of NormalSentiment results with other approaches

In this section, I want to compare the proposed NormalSentiment approach in this thesis, with other approaches in contextual semantic of Sentiment Analysis area.

For this aim, I will refer to the results showed in Saif, H., et al (2015) study and the results are shown in table 6 and 7 below.

In table 6 and 7, The accuracy is calculated and it will be in 2 domains:

1. Subjectivity Classification (subjective vs. objective) Table 6
2. Polarity Classification (positive vs. Negative) Table 7
<table>
<thead>
<tr>
<th>Method</th>
<th>Avg of Subjective &amp; Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
</tr>
<tr>
<td>MPQA Method</td>
<td>63.79</td>
</tr>
<tr>
<td>SentiWordNet Method</td>
<td>63.79</td>
</tr>
<tr>
<td>SentiStrength</td>
<td>62.07</td>
</tr>
<tr>
<td>Senti-Median(SentiWordNet)</td>
<td>81.03</td>
</tr>
<tr>
<td>Senti-Median(MPQA)</td>
<td>77.59</td>
</tr>
<tr>
<td>Senti-Median(Thelwall)</td>
<td>79.31</td>
</tr>
</tbody>
</table>

Table 6: Subjectivity classification Performances (Saif, H, et al 2015)
<table>
<thead>
<tr>
<th>Method</th>
<th>Avg of Polarity Classification (Positive vs. Negative)</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPQA Method</td>
<td></td>
<td>72.50</td>
<td>75.71</td>
<td>68.88</td>
<td>70.63</td>
</tr>
<tr>
<td>SentiWordNet Method</td>
<td></td>
<td>77.50</td>
<td>81.50</td>
<td>81.50</td>
<td>81.50</td>
</tr>
<tr>
<td>SentiStrength</td>
<td></td>
<td>85.00</td>
<td>83.12</td>
<td>86.89</td>
<td>84.00</td>
</tr>
<tr>
<td>Senti-Median(SentiWordNet)</td>
<td></td>
<td>87.50</td>
<td>86.31</td>
<td>84.76</td>
<td>85.45</td>
</tr>
<tr>
<td>Senti-Median(MPQA)</td>
<td></td>
<td>85.00</td>
<td>84.01</td>
<td>80.91</td>
<td>82.14</td>
</tr>
<tr>
<td>Senti-Median(Thelwall)</td>
<td></td>
<td>82.50</td>
<td>80.36</td>
<td>79.06</td>
<td>79.64</td>
</tr>
</tbody>
</table>

Table 7: Polarity Classification (positive vs. negative) Performances (Saif H. et al 2015)

Table 6 and table 7 show the performances in accuracy, Precision, Recall, and F1 for 6 different method in Contextual semantic of Sentiment Analysis Area.

The methods are as following:

- MPQA Method
- SentiWordNet Method
- SentiStrength
- Senti-Median (SentiWordNet)
- Senti-Median (MPQA)
- Senti-Median (Thelwall-Lexicon)
Then we will compare the highest performance measurements among the methods:

1. **Accuracy**: According to tables 6 & 7, the max accuracy belongs to the Senti-Median (SentiWordNet) method with 81% in subjectivity and 87.5% in Polarity.
2. **Precision**: According to tables 6 & 7, the max Precision belongs to the Senti-Median (SentiWordNet) method with 79.45% in subjectivity and 86.31% in Polarity.
3. **Recall**: According to tables 6 & 7, the max Recall belongs to the Senti-Median (SentiWordNet) method with 81.97% in subjectivity and 84.76% in Polarity.
4. **F1**: According to tables 6 & 7, the max accuracy belongs to the Senti-Median (SentiWordNet) method with 80.03% in subjectivity and 85.45% in Polarity.

Therefore, we conclude that among other approaches, Senti-Median(SentiWordNet) excels in accuracy, precision, recall, and F1 which shows that it was the most accurate approach till now.

**7.2.3 Comparison of Performance (NormalSentiment vs. Senti-Median)**

The performance of NormalSentiment is calculated in section 7.2.1 and the excellency of Senti-Median among other approaches is shown in section 7.2.2.

Now, I will be able to compare the performance in table 8 below:

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal sentiment</td>
<td>90%</td>
<td>92.3%</td>
<td>91.1%</td>
</tr>
<tr>
<td>Senticircle</td>
<td>86.31%</td>
<td>84.76%</td>
<td>85.45%</td>
</tr>
</tbody>
</table>

*Table 8: Comparing Sentiment Analysis Result in 2 methods*
According to the table 8, NormalSentiment method surpasses in accuracy from the most accurate approach in sentiment analysis area (Senticircle or Senti Median).

- Precision: NormalSentiment performs better in 90% of precision than Senticircle (86%)
- Recall: NormalSentiment performs better in 92.3% of recall than Senticircle (84.76%)
- F1: NormalSentiment performs better in 91.1% of F-Measure than Senticircle (85.45%)

8. Conclusion

In this thesis work, the claim is that a set of words labeled with their prior emotion is not sufficient for automatic discovery of the emotion of a sentence. The context must also be considered. Based on this claim, I proposed the NormalSentiment method, which is a hybrid method that is both lexiconbased (because it uses different lexicons) and machine learning (ML) based (because it uses AlchemyAPI algorithms).

For the implementation of the proposed method, I analysed sentiment polarity of more than 2000 social media tweets expressing attitudes towards two global cell phone brands (iPhone and Galaxy). Although a single tweet is limited to 140 characters in length, the millions of tweets posted on Twitter almost on a daily basis might provide an unbiased representation of consumers’ sentiment towards services and brands.

I believe that this study contributes to the existing literature in contextual semantics for sentiment analysis for being informed about consumers’ behaviour:

➢ First, the NormalSentiment method can identify a precise contextual interval for the average sentiment of a particular term by using its prior sentiment and co-occurrence degree with other terms.
Second, it can compute the percentage of customers who like/dislike a specific brand.

Third, it can identify whether the opinions of customers about a specific brand are diverse or focused.

Finally, I approached the analysis using the most widely-used microblogging site, Twitter, by employing a mixed methods approach based on both qualitative and quantitative methods. This approach ensures the robustness of our results. Also, it was shown that, because the NormalSentiment method can identify a contextual interval for the average sentiment of the terms, it outperforms baselines method in F-measure. Actually the calculated F-Measure in section 7.2.1 is about 91.1% for the NormalSentiment method which performs better than the most accurate method which was introduced in 2015. Centi-Circle (Senti-Median) was introduced in 2015 with only 85% in F1. But it was proved that NormalSentiment method can surpass in accuracy and F-measure among other method in contextual semantics of sentiment analysis area.

For future work, I plan to expand the NormalSentiment method to be used in tweet-level sentiment analysis. In addition, I plan to study neutral tweets, where datasets are enriched with analogue domain datasets and with different features.

Acknowledgment

At last I want to express my sincere thanks to my dear supervisor, Professor Andrea Ko, and to dear Hassan Saif (Author of Centi-Circle method) for his kind consultation regarding the comparison between NormalSentiment & Senti-Circle(Senti-Media), and also Dr. Majid Eyvazian for their significant and scientific instructions for the revision of this thesis work.
9. References


M. Hu, B. L. (2004). Mining and summarizing customer reviews. the 10th ACM SIGKDD International Conference on Knowledge Discovery and data mining, (pp. 168–177)


